blog からの自動意見抽出をはじめとする多様なアプリケーションを組み込んだオンラインblog分析エンジンの開発

著者	貞光 九月, 乗松 潤矢, 福富 崇博
雑誌名	2006年度CSテクニカルレポート・システム開発型研究プロジェクト特集号
発行年	2006
その他のタイトル	Development of analysis engine employing various applications for extracting opinions from blog
URL	http://hdl.handle.net/2241/104465

blog からの自動意見抽出をはじめとする多様なアプリケーションを組み込んだオンライン blog 分析エンジンの開発

貞 光 九 月[†]　乗 松 潤 矢[††]　福 富 崇 博[††]

近年、膨大な Web のデータから、有益な情報を自動的に抽出する分析エンジンに対する関心が高まっている。我々は、近年の研究成果を組み込んだ新しい分析エンジンを作成し、具体的な研究内容について述べていく。はじめに隠れマルコフモデル (HMM) を用いて文単位で文書構造を捉える評価文書分類、次に、フレーズの尤度を素性として加えた階層的フレーズ機械翻訳システムについて述べ、最後に、blog とオンラインニュース等からクロスワードパズルのヒント文を自動生成するシステムの 3 つについて述べる。

Development of analysis engine employing various applications for extracting opinions from blog

Kugatsu Sadamitsu,[†] Junya norimatsu[††]
and Takahiro Fukutomi [††]

There has been a recent swell of interest in analysis engines which is the automatic extraction system of beneficial information in web. We implemented new analysis engines employing our three developments and introduce each development in this report. In first is sentiment analysis based on hidden Marcov models(HMM) for extracting document structures on intersentence, In second is hierarchical phrase-based translation using another features such as phrase likelihood. In last one is automatic generation system of crossword puzzles hints from weblog and online news.

1. はじめに

近年 blog の登場により、ユーザーが自ら情報を発信することが容易となり、サイトを生産する人々の裾野が広がっている。それと同時に、日々増加している膨大な Web 上の情報を分析し、有用な情報を抽出する分析エンジン[1),2)] に対する要求も高まっている。我々は、blog やオンラインニュースといった Web 上のテキストデータを対象とした分析エンジンの開発を目指している。RSS を配信する blog やオンラインニュースに対象を限ることにより、動的に Web ページを収集する際の運用コストを格段に低く抑えることができる。本稿では昨今の我々の研究成果をアプリケーションとして組み込んだ分析エンジンについて述べる。具体的には、「隠れマルコフモデル (HMM) を用いて文単位で文書構造を捉える評価文書分類」[3)]、「フレーズの尤度を素性として加えた階層的フレーズ機械翻訳システム」、「クロスワードパズルヒント文自動生成システム」という、異なる 3 つの研究について述べ、それらがシステム上で動作することを示す。また、システム上で各アプリケーションの融合が容易になるという利点もある。本稿では、評判文書分類と機械翻訳を組み合わせることにより、外国語の blog からも評判を分析することができる。次節からは、実際のシステムに組み込むそれぞれのアプリケーションについて述べた後、最後にシステムの実際の動作についてのスナップショットを例示する。

† 筑波大学システム情報工学研究科
Graduate School of Systems and Information Engineering, University of Tsukuba
†† 筑波大学 情報学類
Collage of Information Sciences, University of Tsukuba

2. 文書構造を利用した評価文書分類

2.1 評価文書分類の概要

blog のように、ある対象に対する評価を含む文書 (評価文書) を、肯定評価・否定評価の 2 値ラベルに分類する評価文書分類[4)5)] は、その対象に対する評価を定量的に提示できるという点で有益である。本節では従来の単語単位のモデルでは捉えられなかった大域的情報を、単語より大きな文を単位とすることで捉え、評価文書分類の精度向上を図る。具体的には、文を直接の出力シンボルとした HMM を用いて文の連鎖構造を

モデル化していく。しかし、HMM を直接最尤推定した場合、過適応を起こしやすいため、終了状態をモデル内部に表現することや、出力確率に事前分布をおくことでスムージングを試みる。実験では Amazon☆のレビューデータに対して評価文書分類を行い、提案手法の有効性を確認する。

2.2 文を単位とする HMM

2.2.1 文書構造を考慮した評価文書分類法

以下の評価文書は、単純なナイーブベイズ識別[4] を評価文書分類に用いた場合に、分類を誤った評価文書の一例である。

評価文書例

これは今シーズンのヒットです！ティーンエイジャー向きの初心者本かと思ったらさにあらず、ボレロ、マーガレット、アシンメトリーカーデガン、ポンチョ、ケープなど、今年のテイストでシンプルながら簡単すぎない可愛いデザインがいっぱいです。**編み物本って変に懲りすぎておばさんくさいか、簡単すぎて作っても着れないものかどっちかが多いのですが、** これはどれも着れます。 **初心者はガーターとかでマフラー編みがちですが、 あれって単調でつまらないですよね。** 少し凝ったものの方がかえって楽しく編めますよ。 丁寧な説明もついてます。 このお値段でこの内容はお得です。 絶対オススメ！

上記評価文書は、肯定的内容であるにも関わらず、局所的には否定的表現 (太字表記) が多い文書となっている。よって、単語単位でこの文書を評価した場合、否定的評価文書として分類される可能性は高いと言える。分類を誤った評価文書の 8 割以上において、上記例のように本来のラベルとは逆のラベルに現れやすい単語を局所的に含んでいた。しかし、これらの箇所は「評価対象以外の対象に対する批評」や、「他人の経験・言葉の引用」等、評価対象に対するレビュアーの意見そのものを表しているのではないことを考慮しなければならない。単語単位のモデルによってこれを実現することは困難であるが、本節では文単位のモデル化を行うことで、より長距離の情報を取り込むことを試みる。

2.2.2 文単位の HMM による文書構造のモデル化

本節では、各々の文が何かしらの隠れたクラス (例えば「引用」クラスや「異なる対象への評価」クラス) を持ち、そのクラスが遷移していくことで文書構造が成されると仮定する。この仮定において、文自体を出力シンボルとする HMM を用いるのは自然といえる。本節では単語のストリームとして表現された文に対する Moore 型の文単位 HMM を考える。文書 d_k に対し文単位 HMM を用いて付与される確率 P_H を以下のように定義する。

☆ http://www.amazon.co.jp

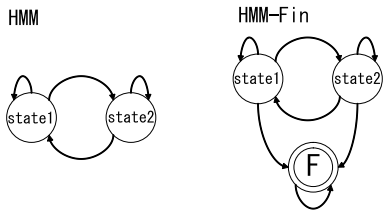

図 1 HMM と終了状態考慮型 HMM(2 状態)

$$P_H(d_k|\boldsymbol{a},\boldsymbol{b}) = \sum_{\boldsymbol{q}_1^{T_k}} \prod_{t=1}^{T_k} a_{q_{t-1}q_t} b_{q_t}(\boldsymbol{s}_{kt}) = \sum_{\boldsymbol{q}_1^{T_k}} \prod_{t=1}^{T_k} a_{q_{t-1}q_t} \prod_{n=1}^{|\boldsymbol{s}_{kt}|} b_{q_t}(w_{ktn}) \quad (1)$$

ここで $\boldsymbol{s}_{kt}$ は t 番目の単語シーケンスを示し、以降「文」とはこの単語シーケンスを指すこととする。t は文書 d_k の文番号、T_k は文書 d_k に含まれる文数、w_{ktn} は文 $\boldsymbol{s}_{kt}$ の n 番目に出現した単語、q_t は t 番目の文が滞在する HMM の状態を示す。また $\boldsymbol{a},\boldsymbol{b}$ はモデルパラメータで、$a_{q_{t-1}q_t}$ は q_{t-1} の状態から q_t へ遷移する確率を表し、$b_{q_t}(w_{ktn})$ は状態 q_t において単語 w_{ktn} を出力する確率を表す。

定義したモデルに対し、EM アルゴリズム (Baum-Welch アルゴリズム) を用いてパラメータ推定を行う。文単位 HMM についての Q 関数は以下のようになる。$\boldsymbol{\theta}$ はモデルパラメータを表す。

$$Q(\boldsymbol{\theta},\boldsymbol{\theta}^{new}) = \sum_k \sum_{\boldsymbol{q}_1^{T_k}} \frac{p(\boldsymbol{q}_1^{T_k}, d_k|\boldsymbol{\theta})}{p(d_k|\boldsymbol{\theta})} \log p(\boldsymbol{q}_1^{T_k}, d_k|\boldsymbol{\theta}^{new})$$

Q 関数をそれぞれのモデルパラメータについて最大化することで、各パラメータの更新式が導出されるが、紙面の都合上ここでは省略する。

2.2.3 終了状態考慮型 HMM

評価文書においては、文書の最後に結論が来ることが多いことから、文書の後側に出現する評価表現は、評点に対して強い影響を与えるという傾向がある[6]。本節ではこの位置情報についても HMM に含めることを試みる。終了状態を表現するため、新たな状態 F を前節で述べた HMM に付加する。文書の最後の文は、必ず状態 F から生成され、また一度状態 F に遷移した後には、他の状態には遷移せず、状態 F に留まり続けると仮定した HMM を、終了状態考慮型 HMM(HMM-Fin) と呼ぶこととする (図 1)。以上の仮定を置くことで、状態 F は各学習ラベルの終了状態に特化した確率分布を持つことが期待される。

2.2.4 文書構造に着目した先行研究

HMM を用いて文書の構造を捉える先行研究として、柴田ら[7] や福井らの研究[8] が挙げられるが、これらはある程度人手によるルールを必要としたり、文の構造

が教師付きデータとして与えられる場合についての検討である。また、文を与件とし、ラベルの条件付確率を直接最大化する CRF(Conditional Random Fields) を用いた評価文書分類[9] が提案されているが、各文毎にクラスのタグを人手で付与する必要があり、コストがかかってしまう。HMM の他に、文単位で文書の構造をとらえる試みとして、RST(Rhetorical Structure Theory)[10] のような人手で定義した木構造を用いる手法もあげられるが、学習コーパスを構成するのにやはり莫大なコストがかかってしまう。それに対し提案手法は、次節で述べるように、文を単なる単語のシーケンスとみなすだけなので、格段に低いコストでモデル学習を行うことが可能である。

2.3 文単位 HMM における出力確率のスムージング

2.3.1 事後分布の期待値を用いたスムージング

2.2.2 節のパラメータ推定法は最尤推定法であるため、学習データに対して過適応しやすく、スムージングの必要が生じる。バイオインフォマティクスの分野において、Brown らは HMM の出力確率 $\boldsymbol{b}$ の事前分布に混合ディリクレ分布を用いたスムージング法を提案している[11]。ここで、単一のディリクレ分布は多項分布の共役事前分布であり、その合成分布は Polya 分布となる。それぞれの確率分布 P_{Dir}, P_{Polya} は以下の式で定義される。

$$P_{Dir}(\boldsymbol{\theta};\boldsymbol{\alpha}) = \frac{\Gamma(\sum_{v=1}^{V}\alpha_v)}{\prod_{v=1}^{V}\Gamma(\alpha_v)} p_1^{\alpha_1-1}\cdots p_V^{\alpha_V-1} \quad (2)$$

$$P_{Polya}(\boldsymbol{s};\boldsymbol{\alpha}) = \frac{\Gamma(\alpha)}{\Gamma(\alpha+y)}\prod_{v}\frac{\Gamma(y_v+\alpha_v)}{\Gamma(\alpha_v)} \quad (3)$$

y_v は文 $\boldsymbol{s}$ 中に単語 v が含まれている数を示し、$y = \sum_v y_v$ である。α_v はモデルパラメータで、$\alpha = \sum_v \alpha_v$ である☆。なお混合ディリクレ分布は、ディリクレ分布の混合分布である。Brown らは学習データとクラスのアラインメントをヒューリスティクスに求めた後、各状態毎に事前分布となる混合ディリクレ分布を推定し、事後分布を再度計算、その期待値をスムージングした出力確率として用いている。本節ではヒューリスティクスにアラインメントをとることはせず、全学習データで学習した単一ディリクレ分布を事前分布として用いることを考える。最終的に、単一ディリクレ分布を事前分布とし、事後分布の期待値を新たな出力確率とする $\boldsymbol{b}^{ex}$ の推定式は以下のようになる。なお、式中 $\gamma_{kt}(i)$ は文書 k 中の文 t が状態 i に滞在する確率であるが、紙面の都合上導出等は省略する。

$$b_i^{ex}(v) = \frac{\sum_k\sum_{t=1}^{T_k}\gamma_{kt}(i)\{y_{ktv}+\alpha_v\}}{\sum_k\sum_{t=1}^{T_k}\sum_{v'}\gamma_{kt}(i)\{y_{ktv'}+\alpha_{v'}\}} \quad (4)$$

☆ 通例に従い、ディリクレ分布のモデルパラメータと HMM における前向き確率には同じ α という変数を用いたが、全く別物である。

しかし、このスムージング法を用いた場合、y_v の値がほとんどの文中の単語について 0 になってしまうため、α_v の重みが相対的に非常に大きくなってしまう。その結果、y の値をほぼ無視した学習となり、事前分布の単一ディリクレ分布そのものが再度モデル化されるという問題が生じた。以上の理由により、本節では評価実験を行っていない。

2.3.2 出力確率に Polya 分布を仮定したスムージング

前節では全文書に通ずる単一ディリクレ分布を事前分布として仮定し、unigram 確率に対するスムージングを施したが、本節では HMM の各状態に Polya 分布を直接仮定することで、新たなモデル (PolyaHMM) を考える。文書 d_k に対し PolyaHMM を用いた場合の確率 P_{PH} は以下のように定式化できる。

$$P_{PH}(d_k|\boldsymbol{a},\boldsymbol{b}) = \sum_{\boldsymbol{q}_1^{T_k}}\prod_{t=1}^{T_k} a_{q_{t-1}q_t} P_{Polya}(\boldsymbol{s}_{kt};\boldsymbol{\alpha}_{q_t}) \quad (5)$$

パラメータ推定については、2.2.2 節と同様に EM アルゴリズムを用いて最尤推定を行うことになる。$\boldsymbol{\alpha}$ についての最終的な更新式は Leaving-One-Out 法[12] を用いて導出される。加えて、PolyaHMM においても、2.2.3 節と同様に終了状態考慮型 HMM を構成することができ、次節の実験においては、双方のモデルについて実験を行う。

2.4 評価文書分類実験と考察

2.4.1 実 験 条 件

本節では評価実験に際し、Amazon からジャンルを問わず 95784 アイテム (商品) に関するレビュー、全 419278 レビューを取得した。Amazon のレビューには評点がレビュアーによって既に付与されており、各評点のレビュー数は、最も低い評点 1 から最も高い評点 5 まで、順に 14224, 15927, 39632, 1033355, 238074 レビューであった。評点 5,4 のレビューを Positive レビュー、評点 1,2 のレビューを Negative レビューとし、それぞれのデータについて各モデルを学習させ、ナイーブベイズ識別を行う。本実験では評点毎に同一数のレビューを用いることとし、学習データは各評点からランダムに 12000 レビューを選択したもののうち、20 単語以下から成るレビューを除外、計 47400 レビューを学習に用いた。1 レビューあたりの平均単語長は 153.71 単語である。テストデータは各評点から 21 単語以上から成るレビューを、学習データ以外からランダムに 100 レビューずつ抽出した。学習・テストそれぞれに含まれるレビューを 10 単語毎に区切り、それぞれを文とした。語彙は全学習データ中に含まれる単語のうち、出現回数が 20 回以上の単語、計 13350 単語である。レビューのタイトル、及びレビュアー名はレビューデータに含めていない。

2.4.2 評価文書分類実験

各モデルにおいて評価文書分類を行った結果を図 2

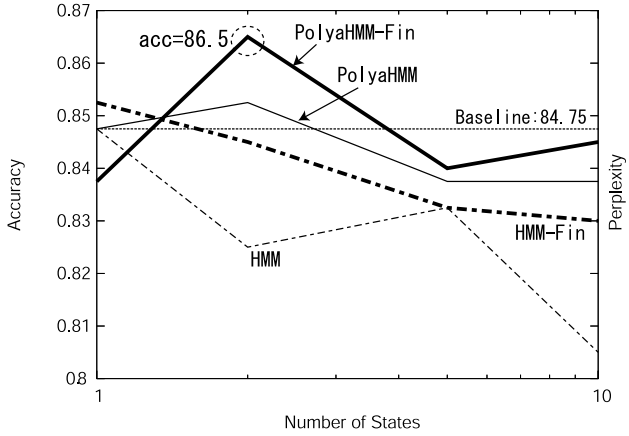

図 2 評価文書分類における正解率

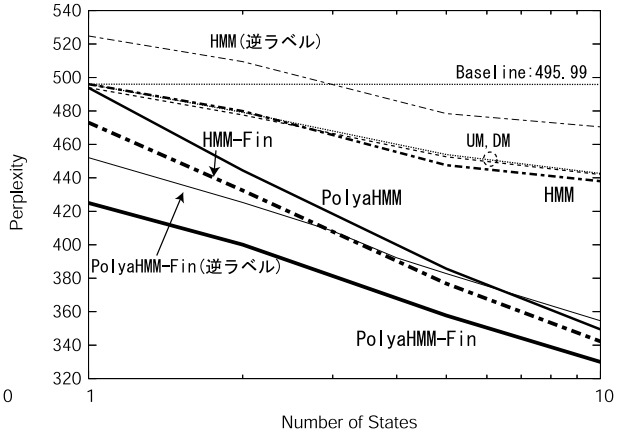

図 3 各モデルのパープレキシティによる比較

に示す。横軸は状態数を表し、1,2,5,10 状態で実験を行った。終了状態考慮型 HMM については、終了状態を除いた状態数を示すこととする。ベースラインはユニグラムモデルによるナイーブベイズ識別である。HMM では 2 状態以上 (1 状態の場合は即ちベースライン) においてベースラインより悪化してしまうものの、HMM-Fin では 1 状態 (終了状態とあわせて 2 状態) の時にベースラインをわずかながら上回っている。これは、終了状態に重みをおいた場合のナイーブベイズ識別を、HMM 内部にモデル化できたことが理由の一つとして考えられる。また、PolyaHMM,PolyaHMM-Fin では、いずれも 2 状態の時に最高精度を示しており、この 2 状態がうまく Positive 状態と Negative 状態をモデル化できている可能性がある。なお、2.2.1 節で挙げた例は、ベースラインで誤り、PolyaHMM-Fin の 2 状態で正解した実際の例である。

しかし PolyaHMM を用いた場合でも、5 状態以上の状態数において分類精度は改善しない。この原因を分析するため、図 3 に各モデルにおけるテストセットパープレキシティの値を示す。ここではテストデータの正解ラベルと同じラベルの学習データを用いた場合のモデルを用いて、パープレキシティを算出している。ただし、HMM と PolyaHMM-Fin については、比較のため正解と逆のラベルで算出したものも示す (図中**逆ラベル**)。また、提案手法から遷移確率を取り除いた場合とみなすことのできる混合モデル 2 種 (Unigram Mixtures[13], 混合ディリクレモデル[12]) との比較もあわせて行っている (図中 UM,DM。横軸は混合数)。

実験結果より、提案手法のいずれについても、状態数を増加させるにつれ、パープレキシティが単調に減少していることが確認できるものの、逆ラベルで学習したモデルに対するパープレキシティも、同様に減少している。この原因として、文単位 HMM の状態数の増加が、文の構造をより正しく捉えていく方向に働くのではなく、文 $\boldsymbol{s}_t$ に含まれるある単語と、それに続く文 $\boldsymbol{s}_{t+1}$ に含まれるある単語との、単語同士の微細な関係に着目した単純な n-gram がモデル化されている可能性が考えられる。これは、長距離 n-gram の間接的なモデル化とも捉えられるが、単語同士の結びつきを表現するだけでは、本節の目的である文書構造のモデル化には至らないため、結果的に逆ラベルのパープレキシティも減少してしまうのではないかと考える。さらに、終了状態考慮型は文書中のあらゆる時点においても、終了状態へ遷移することが可能であるため、単語単位 n-gram に対し、最終状態 F の ユニグラム確率によってスムージングがかかり、結果的にパープレキシティが減少し続けるのではないかと考える。これは PolyaHMM, PolyaHMM-Fin においても、HMM 程直接的ではないにしろ起こりうる原因である。これらの問題を回避するためには、それぞれのモデル間において、確率的な差をいかに大きくしつつモデルを学習していくか課題になると推察される。

2.5 HMM を用いた評価文書分類のまとめ

文単位の HMM によって文の構造を捉えることを提案し、それを用いて評価文書分類の正解精度を上げることを試み、若干ではあるものの改善を示すことができた。今後の課題としては、各ラベル毎の学習データの尤度が高くなるように学習するのではなく、両側のラベルの学習データを考慮しつつ、それぞれの特徴をより明確に捉えられるモデルを生成していくことが課題となる。

また、提案手法は言語モデルとしては優れた性能を示したため、事前分布に対し階層ベイズを用いてスムージングをかける手法[14] や LDA[15] を HMM の各状態とする手法等、HMM に対する他のスムージング法についても試みたい。

3. 階層フレーズを用いた統計的機械翻訳

3.1 統計的機械翻訳の概要

統計的機械翻訳は機械翻訳に統計的手法を取り入れた手法である[?]。統計的機械翻訳では、ベイズの定理に基づいた以下の式によって翻訳がなされる。ここで F は翻訳される元の言語 (原言語)、E は翻訳される先の言語 (目的言語) と呼ばれる。

$$\begin{aligned}\hat{E} &= \arg\max_{E} P(E|F) \\ &= \arg\max_{E} P(E)P(F|E)\end{aligned} \tag{6}$$

この式より、統計的機械翻訳システムには 3 種類の研究分野があると言える。

- 言語モデル $P(E)$:ある文 E がその属する言語の文として生起する確率を与える。
- 翻訳モデル $P(F|E)$:ある文 E が原言語の文 F に翻訳される確率を与える。
- デコーダ $\arg\max_E$:最も翻訳文としての確率が高くなる目的言語の文 E を返す。

近年、統計的機械翻訳は単語単位での翻訳からフレーズ単位の翻訳、階層的フレーズ翻訳へと発展しており、以下の節ではこれらの手法について述べる。

3.2 フレーズを用いた統計的機械翻訳

フレーズモデル (Phrase-based model) はフレーズを翻訳の単位とするモデルである。フレーズモデルを用いた翻訳は以下のように定式化される[16)17)]。原言語文 F と目的言語文 E が I 個のフレーズ対 $\hat{f}_1^I$、$\hat{e}_1^I$ に翻訳されるとすると、翻訳文 $\hat{E}$ は、

$$\hat{E} = \arg\max_{E} P\left(\hat{f}_1^I|\hat{e}_1^I\right). \tag{7}$$

このとき、$\hat{f}_i$、$\hat{e}_i$ をそれぞれ原言語、目的言語の i 番目のフレーズであるとして、

$$P\left(\hat{f}_1^I|\hat{e}_1^I\right) = \prod_{i=1}^{I} \phi\left(\hat{f}_i|\hat{e}_i\right) d\left(a_i - b_{i-1}\right) P_w\left(\hat{f}_i|\hat{e}_i, a\right)^{\lambda} \tag{8}$$

と仮定する。ここで、a_i は、目的言語のフレーズ $\hat{e}_i$ に対応する原言語のフレーズの最初の単語位置、b_{i-1} は、目的言語のフレーズ $\hat{e}_{i-1}$ に対応する原言語のフレーズの最後の単語位置であるとする (図 4)。λ は適当な重みである。このとき、フレーズ翻訳確率 ϕ、歪み確率 d、語彙重み P_w を式 9 から式 13 によりそれぞれ定義する。式中 $count(x, y)$ は任意のペア x, y がコーパスから抽出された回数である。

$$\phi\left(\hat{f}_i|\hat{e}_i\right) = \frac{\text{count}\left(\hat{f}_i, \hat{e}_i\right)}{\sum_{\hat{f}} \text{count}\left(\hat{f}_i, \hat{e}_i\right)} \tag{9}$$

α を適当な数として、

$$d\left(a_i - b_{i-1}\right) = \alpha^{|a_i - b_{i-1} - 1|} \tag{10}$$

また、lexical weight $P_w(\hat{f}|\hat{e})$ は、以下の語彙翻訳確率分布 $w(f|e)$、単語アラインメント付き語彙重み

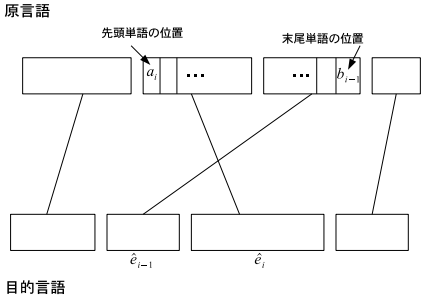

図 4 フレーズ歪みのパラメータ a_i,b_{i-1} の例

$P_w(\hat{f}|\hat{e}, a)$ を用いて計算する。

$$w\left(f|e\right) = \frac{\text{count}\left(f, e\right)}{\sum_{f'} \text{count}\left(f', e\right)}. \tag{11}$$

$$P_w\left(\hat{f}|\hat{e}, a\right) = \prod_{i=1}^{n} \frac{1}{|\{j|(i,j) \in a\}|} \sum_{\forall(i,j)\in a} w\left(f_i, e_j\right). \tag{12}$$

すなわち、

$$P_w\left(\hat{f}|\hat{e}\right) = \max_{a} P_w\left(\hat{f}|\hat{e}_i, a\right)^{\lambda}. \tag{13}$$

3.3 階層フレーズを用いた統計的機械翻訳

フレーズを階層的にとらえる事でフレーズのペアを CFG の対 (Synchronous-CFG) として表現したものを階層フレーズモデル (hierarchical phrase-based model) と呼ぶ[18)]。この階層フレーズは、非終端記号として X と S のみを認める。X は実例からの学習によって得られた階層フレーズ対であり、S は大域的なフレーズである。以下、S を用いた階層フレーズ対を S ルールと呼ぶ。S ルールとして $S \rightarrow < X_1, X_1 >$、$S \rightarrow < S_1X_2, S_1X_2 >$ を認める。ここで、X_n、S_n は階層フレーズ S、X の対がそれぞれの n に関して対応関係があることを表すための略記であるとする。これを用いて原言語を構文解析することで、対応する目的言語の文が生成される。

得られた階層フレーズに、log-linear モデルで以下のように重みを与える。

$$w\left(X \rightarrow < \gamma, \alpha >\right) = \prod_{i} \phi_i\left(X \rightarrow < \gamma, \alpha >\right)^{\lambda_i}. \tag{14}$$

ここで、ϕ_i はそれぞれ以下の要素に対応し、λ_i は適当な重みである。

- フレーズ翻訳確率 $P\left(\alpha|\gamma\right)$、$P\left(\gamma|\alpha\right)$
- lexical weight $P_w\left(\gamma|\alpha\right)$、$P_w\left(\alpha|\gamma\right)$
- フレーズペナルティー $\exp(1)$

また、S ルールに関しては、以下の重みを与える[18)]。

$$w\left(S \rightarrow < S_1X_1, S_1, X_2 >\right) = \exp\left(-\lambda\right) \tag{15}$$

ここで、λ は適当な重みである。$S \rightarrow < X_1, X_1 >$ には重みを与えない。D を目的言語の文、原言語の文、構文木対の組であるとすると、翻訳候補の重みを以下

のように定める[18]。

$$w(D) = \prod_{(r,i,j)\in D} w(r)\, P_{LM}\left(\boldsymbol{e}'\right)^{\lambda_{LM}} \exp\left(-\lambda_{wp}|\boldsymbol{e}'|\right) \tag{16}$$

ここで、i、j は、対応する r によりカバーされる原言語文の範囲を示す。また、$\boldsymbol{e}'$ は f_i^j に対応する目的言語文の部分単語列である。デコーダはこの重み $w(D)$ を最大にする目的言語の文 $\hat{\boldsymbol{e}}$ を求めれば良い。

3.4 階層フレーズモデルの検討

従来手法においては階層フレーズの重みを決めるための要素として、フレーズの翻訳確率や、lexical weight などを用いている。しかし、確率的構文解析を行う際に用いられている階層フレーズ自体の尤度が従来のモデルでは考慮されておらず、これを利用することで翻訳精度向上につながるのではないかと考えた。すなわち、確率的な構文解析を行う階層フレーズモデルにおいて、階層フレーズ自身の確率を用いることで、より精度の高い構文解析ができると考える。そこで本稿では、新たに $P(\hat{f})$、$P(\hat{e})$、$P(\hat{f},\hat{e})$ を要素として加えた。それぞれの要素の推定式は以下のようになる。

$$P(\hat{e}) = \frac{\mathrm{count}(\hat{e})}{\sum_{\hat{e}'} \mathrm{count}\left(\hat{e}'\right)} \tag{17}$$

$$P\left(\hat{f}\right) = \frac{\mathrm{count}\left(\hat{f}\right)}{\sum_{\hat{f}'} \mathrm{count}\left(\hat{f}'\right)} \tag{18}$$

$$P\left(\hat{f},\hat{e}\right) = \frac{\mathrm{count}\left(\hat{f},\hat{e}\right)}{\sum_{(\hat{f}',\hat{e}')} \mathrm{count}\left(\hat{f}',\hat{e}'\right)}. \tag{19}$$

ここで、

$$\mathrm{count}\left(\hat{f}\right) = \sum_{\hat{e}} \mathrm{count}\left(\hat{f},\hat{e}\right) \tag{20}$$

$$\mathrm{count}(\hat{e}) = \sum_{\hat{f}} \mathrm{count}\left(\hat{f},\hat{e}\right) \tag{21}$$

である。

次に、これらの値をそれぞれスムージングする。フレーズモデルや階層フレーズモデルの要素として翻訳確率 $P(\hat{f}|\hat{e})$ が用いられている。$\hat{f}'$ をコーパス中 1 回しか出現しなかった原言語フレーズであるとした場合、対応する目的言語フレーズ $\hat{e}'$ も 1 つしか存在しないため、$P(\hat{e}'|\hat{f}') = 1$ になってしまう。これは実際には複数通りあるかもしれない翻訳結果に大きな影響を与えてしまう可能性があり、現実的なモデルとしての確率を与えられているとは言い難い。フレーズモデルにおいては、確率値に対してスムージングを行うと翻訳精度が向上するという報告がなされている[19]。言語モデルのスムージングと同様に翻訳確率をスムージングするこの手法を階層フレーズモデルに適応する。

今回は言語モデルにおいて用いられる Absolute Discounting を翻訳確率に応用する。$P\left(\hat{e}|\hat{f}\right)$ は以下のように変形できる。

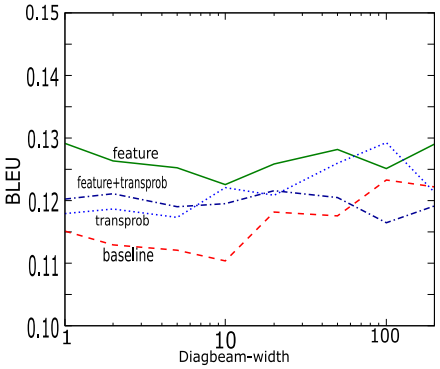

図 5 実験結果 (stack=100)

$$P\left(\hat{e}|\hat{f}\right) = \frac{P\left(\hat{e},\hat{f}\right)}{P\left(\hat{f}\right)} = \frac{\mathrm{count}\left(\hat{e},\hat{f}\right)}{\mathrm{count}\left(\hat{f}\right)} \tag{22}$$

ここから定数 δ だけ共起回数を減じればよいので、Absolute Discounting した確率値は以下のようになる。

$$P_{ad}\left(\hat{e}|\hat{f}\right) = \frac{\mathrm{count}\left(\hat{e},\hat{f}\right) - \delta}{\mathrm{count}\left(\hat{f}\right)} \tag{23}$$

この時、未知語に対する確率は付与しないこととする。$P\left(\hat{f}|\hat{e}\right)$ も同様にスムージングを行う。また、新しく加えたフレーズの尤度についても同様に Absolute Discounting を行う。

3.5 階層フレーズを用いた統計的機械翻訳の実験

実験には読売新聞-The Daily Yomiuri 対訳コーパス (1989 年〜2005 年) を用いた。development テストセットには対訳精度[20] 上位 10000 対からランダムに 1000 対とり、そこからさらにランダムに 100 対とって用いた。open テストセットは development テストセットをとった残りの 9000 対からランダムに 1000 対とり、そこからさらに 100 対とって用いた。学習コーパスには対訳精度上位 10001 番目から 25 万対用いた。言語モデルには SRI Language Modeling Toolkit の Modified Kneser-Ney Discounting を用いて作成した trigram モデルを用いた。非終端記号を含まない階層フレーズについてはフレーズベースモデルで用いられるフレーズテーブルを用いた。

実験結果を図 5 に示す。図中、フレーズ尤度を加えた結果を feature、スムージングを行った結果を transprob、フレーズ尤度を加え、かつ翻訳確率とフレーズ尤度の双方にスムージングを施した結果を feature+transprob と記している。 実験結果より、提案手法のいずれの手法においても BLEU 値の向上が見られた。しかし feature+transprob においては、それぞれの手法を単独で用いた場合ほどの向上が見られなかった。この原因の分析については、今後の研究課

題である。

4. クロスワードパズルにおけるヒント文自動生成

4.1 クロスワードパズルの概要

クロスワードパズルは広く一般に普及しているパズルであるが、人手で作成する際には多大な労力を要する。クロスワードの生成では盤面の生成[21),22)]とヒント文の生成[23)]という2つの生成段階に分けられるが、本節では特にヒント文の自動生成手法について述べる。辞書や新聞等のコーパスを用いることで、定義文等を利用したヒント文を作成することは可能であるが、データが静的であるために、ヒント文の数がある程度限られてしまう。そこで本節では、常に新しいテキストデータを取得できるblog及びオンラインニュースを利用した多様なヒント文の自動生成を試みる。

4.2 オンラインニュースからの虫食いヒント文生成

ヒント文に答えとなる単語(キーワード)が含まれる文から、キーワードを伏せることで提示されるヒント文を虫食いヒント文と呼ぶ。どのようなテキストデータに対しても、一定のルールを用いることで虫食いヒント文を生成するのは容易であるが、周囲に出現する単語とキーワードとの関連性が低い場合、キーワードの類推が困難で、ヒント文として適さない可能性が高い。

そこで、我々はニュースの見出しを用いて虫食いヒント文を生成することとした。ニュースの見出しに含まれる単語は互いに深い関連性を持つと考えられる。例えば「第85回天皇杯サッカー・浦和レッズ25年ぶり優勝」という見出しでは、「天皇杯」「サッカー」「浦和レッズ」「優勝」の間には明らかな関連性があり、このうち1単語を伏せたとしても、他の単語からの推測は可能である。このような見出し文は、虫食いヒント文として適しているといえる。

4.3 相互情報量を用いたblogからのヒント文生成

前節でも述べた通り、blogは多様な種類のテキストが混在しているため、単純に虫食い問題を作るだけではヒント文として適さない場合が往々にして存在する。そこで本節では相互情報量を用いることで、キーワードと関連性の高い単語のみに着目してヒント文を生成する手法を提案する。本節で用いる相互情報量は以下で定義される。

$$I(x,y) = \log \frac{N \cdot df(x,y)}{df(x)df(y)} \tag{24}$$

ここで、Nは学習コーパスに含まれる総文数、$df(x,y)$は単語x,yを共に含む文数、$df(x), df(y)$はそれぞれx,yを含む文数である。

キーワード「川」に対し「ナイル、アマゾン、信濃」といったように、単語を列挙することでヒント文を生成する場合には、単純に相互情報量の大きい単語を選択する。また単語ではなく、文自体をキーワードを連想させるヒント文とすることもできる。例えばキーワードが「通過」の場合、「この駅では、特急は止まりません」のように、関連単語(「駅」「特急」「止まる」)を多く含む文をヒント文として提示することとなる。具体的には、文に含まれる単語の相互情報量の算術平均値が大きいものを複数個列挙することでヒント文とする。

4.4 クロスワードパズルヒント文生成実験

4.4.1 オンラインニュースからのヒント文生成実験

実験データにウィキニュースから取得した見出しデータ☆を用い、ヒント文の生成実験を行った。得られた見出し数は1762個である。本実験ではランダムに選択した100個のキーワードに対するヒント文の生成を試みた。その結果、重複無しで23文、重複ありで73文のヒント文が生成できた。

次に、ヒント文としての妥当性を確認するため、生成されたヒント文を、人手によって「有用」「有用でない」「どちらともいえない」の3パターンに分類した。その結果、付与されたヒント文全73文のうち51文が「有用」、15文が「有用でない」、7文が「どちらともいえない」に分類された。生成された実際のヒント文の例を表1に示す。

表1 オンラインニュース見出しによるヒント文生成結果

	ヒント文	キーワード
成功例 (1)	○○○○イーグルス・田尾監督を解任	楽天
成功例 (2)	加藤紘一元自民○○○○長自宅全焼	幹事
失敗例 (1)	2005年○○○○○議会選挙最終結果	ポーランド

4.4.2 blogからのヒント文生成実験

Excite ブログ☆☆から取得したデータ計127456記事を用い、ヒント文の生成実験を行った。相互情報量の高い単語を列挙したヒント文の例を表2に、相互情報量の高い単語を多く含む文を列挙したヒント文とした例を表3示す。

表2 単語の列挙によるヒント文生成結果

	ヒント文	キーワード
成功例 (1)	急ぐ・寄り道・夕焼け	帰り道
成功例 (2)	永遠・永久・ちとせ	千代
成功例 (3)	四日市・紀伊本線・伊賀	三重

表3 文の列挙によるヒント文生成結果

	ヒント文	キーワード
成功例 (1)	・蝉が鳴いている。 ・ちらほらと聞こえてくる。	声
成功例 (2)	・コツコツコツコツ… ・今日は統計学と関数解析。	復習

☆ http: //download.wikimedia.org/jawikinews/20061222/
☆☆ http://www.exblog.jp/

図 6 評価文書分類システムのスナップショット

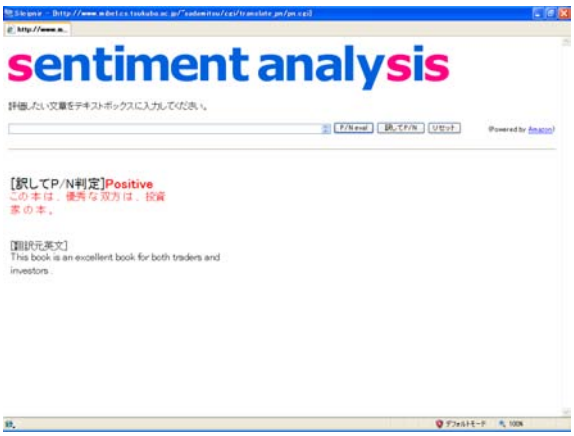

図 7 翻訳システム (訳して P/N) のスナップショット

5. 分析システムの実装

5.1 評価文書分類システムの実装

はじめに評価文書分類を CGI で実装したシステムについて述べる。図 6 は実際の動作のスナップショットである。HMM モデルには PolyaHMM-Fin の 2 状態、それぞれのモデルの学習データは 2.4.2 節で述べたものと同じデータを用いている。ユーザーはテキストボックスに評価したい文書を入力した後、「P/N eval」ボタンを押下することで、評価文書分類結果と、それぞれの文が肯定モデル・否定モデルのどちらからより高い確率を付与されたかをカラーで表示する。画面上、左から提案手法である HMM、ナイーブベイズ、ナイーブベイズの詳細 (各単語毎の、ナイーブベイズにおけるユニグラム比をとり、その比が逆のモデルに対して明らかに大きい単語を、フォントサイズで強調している) である。図 6 では、2.2.1 節で示した例についての動作を確認している。

5.2 翻訳システムの実装

5.2.1 翻訳システム-訳して P/N

次に、3.4 節で述べた、フレーズ尤度を要素として追加した翻訳システムの実装について述べる。学習コーパスなどの条件は 3.5 節に示したものと同様である。図 7 は前節で用いたスナップショットと同じ画面であるが、英語文を入力とし、「訳して P/N」ボタンを押下することで、翻訳結果と、翻訳結果からの評価文書分類結果が表示される。ここでは比較的うまく翻訳できた一例☆を示しているが、実際には翻訳システムの学習データが新聞データであることから、スタイルやジャンルが大きく異なる blog を入力とした場合、新聞を入力とした場合よりも概ね精度の低い結果となった。なお、評価文書分類手法には、PolyaHMM-Fin の 2 状態を用いている。

5.2.2 RSS 翻訳システム-訳して RSS

前節で述べた翻訳システムを応用し、海外のオンラインニュースサイトで配信されている RSS を日本語に翻訳するシステムを作成した。対象をニュースサイトに限定した理由は、前節でも述べたように、翻訳システムの学習データのスタイルやジャンルに似た入力の方が、翻訳精度が良いためである。図 8 は英語版 wikinews(2/7 付) ☆☆から取得した RSS を、システムによって翻訳、RSS を再度作成し、それを Mozilla Thunderbird で読み込んだ際のスナップショットである。画面上部の受信した RSS の件名部がタイトル翻訳結果を示し、画面下部が翻訳元英文を示している。

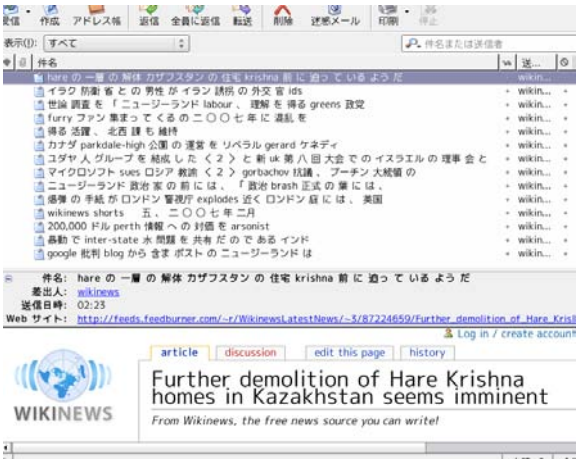

図 8 RSS 翻訳システム (訳して RSS) によるタイトル翻訳

5.3 クロスワードヒント文自動生成システムの実装

最後にクロスワードヒント文の実装について述べる。

☆ "Value Blog Review" (http://valueblogreview.blogspot.com) 中の書籍に対するレビューの一部

☆☆ http://meta.wikimedia.org/wiki/Wikinews

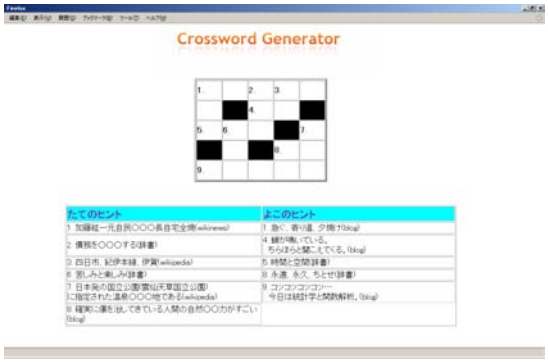

図 9 クロスワードヒント文自動生成システムのスクリーンショット

本稿では盤面生成とキーワード選択については考察していないため、あらかじめ用意した盤面とキーワードに対し、ヒント文のみを生成することとする。図 9 はシステムが生成したヒント文の例である。blog とオンラインニュースだけでは多様性が乏しいため、ヒント文には辞書☆や wikipedia☆☆等の定義文もあわせて用いている。

6. おわりに

本論文では、我々の行ってきた研究成果を実装した分析システムについて述べた。今後の課題は、blog 及びオンラインニュースを動的にデータベースとして保持し、実際にオンライン上で動作させることである。また、本稿で実装した各アプリケーションの学習データは、レビューや新聞等、分析対象である blog とは異なっているため、それぞれの研究分野において、より汎用性の高いシステムが要求される。さらに、blog から意見を抽出する方向へ特化した考察を加えていくことも必要である。文書構造を捉えるという今回の試みは、評価文書分類にとどまらず、文書の要約や、対象に対する要望の抽出等、様々なアプリケーションに応用できると考えている。

謝　辞

本研究の一部は，魅力ある大学院教育イニシアティブ「実践 IT 力を備えた高度情報学人材育成プログラム」による．

参 考 文 献

1) Cass, S.: Fountain of knowledge, *IEEE Spectrum*, Vol. 41, No. 1, pp. 68–75 (2004).
2) Nanno, T., Suzuki, Y., Fujiki, T. and Okumura, M.: Automatically Collecting, Monitoring, and Mining Japanese Weblogs, *Proceeding of WWW2004: the 13th international World Wide Web conference* (2004).
3) 貞光九月, 山本幹雄: 文を単位とする文書構造を用いた評価文書分類, 言語処理学会第 13 回年次大会 (2007). To appear.
4) Pang, B. and Lee, L.: Thumbs up? Sentiment Classification using Machine Learning Techniques, *Proceedings of the Conference on Empirical Methods in Natural Language processing(EMNLP)*, pp. 76–86 (2002).
5) 乾孝司, 奥村学: テキストを対象とした評価情報の分析に関する研究動向, 自然言語処理学会論文誌, Vol. 13, No. 3, pp. 201–241 (2006).
6) Taboada, M. and Grieve, J.: Analyzing appraisal automatically, *AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications* (2004).
7) 柴田知秀, 黒橋禎夫: 隠れマルコフモデルによるトピックの遷移を捉えた談話構造解析, 言語処理学会第 11 回年次大会, pp. 109–112 (2005).
8) 福井義和, 北研二, 永田昌明, 森元逞: 確率・統計的手法による対話構造のモデル化, 情報処理学会研究報告,NL-112, pp. 111–118 (1996).
9) Mao, Y. and Guy, L.: Isotonic Conditional Random Fields and Local Sentiment Flow, *Neural Information Processing Systems*, Vol.18 (2007).
10) Mann, W. C. and Thompson, S. A.: Rhetorical Structure Theory: Description and Construction of Text Structures, *ISI Technical Report* (1986).
11) Brown, M., Hughey, R., Krogh, A., Mian, I., Sjolander, K. and Haussler, D.: Using Dirichlet Mixture Priors to Derive Hidden Markov Models for Protein Families, *Intelligent Systems for Morecular Biology* (1993).
12) 貞光九月, 三品拓也, 山本幹雄: 混合ディレクレ分布を用いたトピックに基づく言語モデル, 電子情報通信学会論文誌 D-II, Vol. J88, pp. 1771–1779 (2005).
13) Iyer, R. and Ostendorf, M.: Modeling Long Distance Dependence in Language: Topic Mixtures vs. Dynamic Cache Models, *Proc. ICSLP '96*, Vol. 1, Philadelphia, PA, pp. 236–239 (1996).
14) 貞光九月: 階層ベイズモデルを用いた混合ディリクレモデルのスムージング法, 筑波大学システム情報工学研究科修士論文 (2006).
15) Blei, D. M., Ng, A. Y. and Jordan, M. I.: Latent Dirichlet Allocation, *Neural Information Processing Systems*, Vol. 14 (2001).
16) Koehn, P., Och, F. J. and Marcu, D.: Statistical phrase-based translation, *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association*

☆ 学研現代新国語辞典 (学研))
☆☆ http://ja.wikipedia.org/wiki/

for Computational Linguistics on Human Language Technology, Morristown, NJ, USA, Association for Computational Linguistics, pp. 48–54 (2003).
17) Koehn, P.: *Pharaoh a Beam Search Decoder for Phrase-Based Statistical Machine Translation Models User Manual and Description for Version 1 2* (2004).
18) Chiang, D.: Hierarchical phrase-based translation, *Computational Linguistics*, Vol. 33, No. 2 (2007). To appear.
19) Foster, G., Kuhn, R. and Johnson, H.: Phrasetable Smoothing for Statistical Machine Translation, *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, Sydney, Australia, Association for Computational Linguistics, pp. 53–61 (2006).
20) Utiyama, M. and Isahara, H.: Reliable Measures for Aligning Japanese-English News Articles and Sentences, *ACL-2003*, pp. 72–79 (2003).
21) Berghel, H.and Yi, C.: Crossword compiler compilation, *The Computer Journal 30*, pp. 276–280 (1989).
22) 加部通明, 生方俊典: クロスワードパズルの遺伝的アルゴリズムによる作成, 第 52 回平成 8 年前期情報処理学会全国大会講演論文集.
23) Aherne, A. and Vogel, C.: Crossing WordNet with Crosswords,Netting Enhanced Automatic Crossword Generation, *Trinity College technical report* (2005).